Slobodan Petrović, Amparo Fúster and Raúl Durán
Dpto. Tratamiento de la Información y Codificación, C.S.I.C.,
Serrano 144, 28006 Madrid, Spain
e-mail: {slobodan, amparo}@iec.csic.es, raul@tic.iec.csic.es

ABSTRACT

Pseudorandom generator schemes containing irregularly clocked linear feedback shift-registers (LFSRs) have become popular because of the properties of their output sequences (long period and large linear complexity). In this paper, a cryptanalytic attack on such schemes that utilizes the divide-and-conquer paradigm is presented. The general statistical model of such generators is given. The appropriate family of edit-distance measures is defined and the flow of an attack on some particular schemes is described. The time and space complexities of the attack are discussed.

Keywords: Cryptanalysis, Edit-distance, Clock-controlled sequence, Ciphertext-only attack, Statistical model.

1. INTRODUCTION

Pseudorandom sequence generators are often used in practical data protection systems because of ease of their implementation, low-cost key distribution and good practical secrecy. In order for such generators to be used in data protection systems, their output sequences must satisfy some predefined criteria, the most important of which are long period, large linear complexity and good statistical properties. The generators that include irregularly clocked linear feedback shift registers (LFSRs) as building blocks satisfy those criteria easily. In this paper, a cryptanalytic attack on this family of generators is presented.

The general pseudorandom generator scheme that is investigated here consists of n binary LFSRs of lengths $l_1, l_2, \ldots, l_n$, whose outputs are combined in a function $\mathfrak{F} = (\mathfrak{f}, \mathfrak{g})$ with $\mathfrak{L}$ bits of memory, where $\mathfrak{f} : \{0,1\}^{\mathfrak{L}+n} \longrightarrow \{0,1\}$ is the output function and $\mathfrak{g} : \{0,1\}^{\mathfrak{L}+n} \longrightarrow \{0,1\}^{\mathfrak{L}}$ is the next-state function of $\mathfrak{F}$. Each LFSR is irregularly clocked by a distinct subgenerator.

A statistical model of this scheme is presented in the Fig. 1. The n input sequences, $X_1, X_2, \ldots, X_n$ of lengths $N_1, N_2, \ldots, N_n$, respectively are transformed into the sequence Y of length M by means of their corresponding decimation sequences $d_i, i = 1, \ldots, n$, the function $\mathfrak{F}$ and the noise sequence Z in which the probability of one is $P(z_i = 1) = p < 0.5$, $i = 1, \ldots, M$.

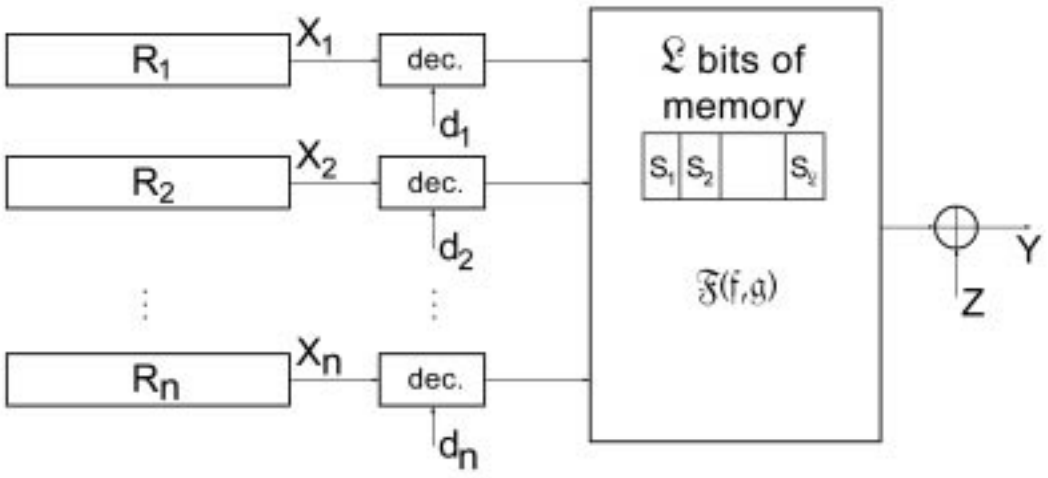

Fig. 1 - The statistical model of the generator containing irregularly clocked LFSRs

The function $\mathfrak{F}$ can generally be decomposed into two parts [2]:

1. a balanced part $F(f,g)$, where $f : \{0,1\}^{L+K} \longrightarrow \{0,1\}$, and $g : \{0,1\}^{L+K} \longrightarrow \{0,1\}^{L}$, f being balanced;
2. a non-balanced (so called noisy) part $F'(f',g')$, where $f' : \{0,1\}^{L'+n-K} \longrightarrow \{0,1\}$, and $g' : \{0,1\}^{L'+n-K} \longrightarrow \{0,1\}^{L'}$, f' being non-balanced;

where $\mathfrak{L} = L + L'$.

The set of sequences $\{X_1, X_2, \ldots, X_n\}$ is related to the sequence Y through its ordered subsets $\{X_{j_1}, X_{j_2}, \ldots, X_{j_K}\}$, generated by the LFSRs $\mathrm{R}_{j_1}, \mathrm{R}_{j_2}, \ldots, \mathrm{R}_{j_K}$, respectively, $j_i \in \{1, \ldots, n\}$. The particular subset depends on the partition of $\mathfrak{F}$. Given one of such ordered subsets, the corresponding constrained edit-distance measure is defined as the minimum number of elementary edit-operations (deletions and substitutions) needed to transform this subset into the sequence Y. The maximum length E_i of a run of deletions in the sequence X_{j_i} represents the constraint, $i = 1, \ldots, K$.

2. CONSTRAINED EDIT-DISTANCES

Let $[a_1], \ldots, [a_K]$ and $[b]$ be $K+1$ sequences of lengths $N_1, \ldots, N_K$ and M ($M \leq \min\{N_1, \ldots, N_K\}$), respectively, over the alphabet $\{0,1\}$. Let $f : \{0,1\}^{L+K} \to \{0,1\}$ be the output function and let $g : \{0,1\}^{L+K} \to \{0,1\}^{L}$ be the next-state function of the function with memory $F = (f,g)$. Let S be any state of the memory of the function F.

Consider the problem of transforming the ordered set $([a_1], \ldots, [a_K])$ into $[b]$ by using deletions of symbols in $[a_1], \ldots, [a_K]$ respectively reducing them to the sequences $[a'_1], \ldots, [a'_K]$ of the same length M, and then substitutions for symbols in the sequence $[c]$ obtained from $[a'_1], \ldots, [a'_K]$ by the function $F = (f,g)$, that is, at the time instant t:

$$S_{t+1} = g(S_t, a'_{1t}, \ldots, a'_{Kt}), \qquad 1 \leq t \leq M-1 \quad (1)$$

$$c_t = f(S_t, a'_{1t}, \ldots, a'_{Kt}), \qquad 1 \leq t \leq M, \quad (2)$$

starting from a given initial state S_1.

Constrained edit-distance (CED) D_F between the ordered set $([a_1], \ldots, [a_K])$ and $[b]$ is then defined as the minimum sum of elementary edit-distances associated with the edit-operations of deletion and substitution needed to transform $([a_1], \ldots, [a_K])$ to $[b]$ subject to the constraint that the maximum number of consecutive deletions in the sequence $[a_i]$ is E_i, $1 \leq i \leq K$. It follows that

$$M \leq N_i \leq M + E_i(M+1), \; 1 \leq i \leq K. \quad (3)$$

Nonnegative real-valued elementary distances are defined by:

1. $d(x,\phi)$ is the elementary distance associated with deletion of $x \in \{0,1\}$ from the sequence $[a_i]$, $1 \leq i \leq K$, where the 'empty' symbol ϕ is introduced to represent deletion;
2. $d(x,y)$ is the elementary distance associated with substitution of x by y, $x, y \in \{0,1\}$.

In order to define the explicit expression for the constrained edit-distance, we represent an edit-transformation sequentially. Namely, we define a $(K+1)$-dimensional edit-sequence $\mathcal{E} = ([\alpha_1], \ldots, [\alpha_K], [\beta])$ over the alphabet $\{0,1,\phi\}$ of length $\mathcal{L} = M(1-K) + \sum_{i=1}^{K} N_i$ by the following encoding scheme. First, let for an arbitrary finite length sequence $[a]$ over $\{0,1\}$, $[\alpha]$ denote any finite length sequence over $\{0,1,\phi\}$ such that by removing all the 'empty' symbols from $[\alpha]$ one obtains $[a]$. Then, an edit-sequence $([\alpha_1], \ldots, [\alpha_K], [\beta])$ is defined by the following rules:

1. The lengths of $[\alpha_1], \ldots, [\alpha_K]$ and $[\beta]$ are all equal to $\mathcal{L} = M(1-K) + \sum_{i=1}^{K} N_i$, which is the total number of deletions and substitutions.
2. If $\alpha_1(i), \ldots, \alpha_K(i)$ and $\beta(i)$ are all non-empty symbols, then the substitution of the corresponding symbol in the combination sequence $[c]$ by the symbol $\beta(i)$ takes place, for any $1 \leq i \leq \mathcal{L}$.
3. If $\beta(i)$ and $K-1$ symbols of $\alpha_1(i), \ldots, \alpha_K(i)$ are all 'empty' symbols and one of $\alpha_1(i), \ldots, \alpha_K(i)$ is not the 'empty' symbol, then the deletion of that symbol takes place, for any $1 \leq i \leq \mathcal{L}$.
4. For any $1 \leq i \leq \mathcal{L}$ no other cases apart from 2. and 3. are allowed.
5. In any sequence of consecutive deletions, one first deletes symbols from $[a_1]$, then from $[a_2]$, and so on.
6. The maximum number of consecutive deletions in $[a_i]$ is $E_i, 1 \leq i \leq K$.

The rule 5. ensures the unique sequential representation of edit-transformations, meaning that there is an one-to-one correspondence between the set of all the permitted edit-sequences $([\alpha_1], \ldots, [\alpha_K], [\beta])$ defined as above, denoted by , $([a_1], \ldots, [a_K]; [b])$, and the set of all the permitted edit-transformations of $([a_1], \ldots, [a_K])$ into $[b]$.

For each edit-sequence $\mathcal{E}$ one can compute the memory state sequence $[S(i)]_{i=1}^{M}$ using the recursion (1), and then form the extended state sequence $[\Sigma(i)]_{i=1}^{\mathcal{L}}$ over $\{0,1,\phi\}$ inserting the 'empty' symbol wherever there is the 'empty' symbol in $\mathcal{E}$.

The constrained edit-distance can be expressed in terms of edit-sequences and extended state sequences by:

$$D_F([a_1],\ldots,[a_K];[b]) =$$

$$= \min\left\{\sum_{i=1}^{\mathcal{L}} d_F(\Sigma(i),\alpha_1(i),\ldots,\alpha_K(i);\beta(i)) \mid \right.$$

$$\left.([\alpha_1],\ldots,[\alpha_K],[\beta]) \in , ([a_1],\ldots,[a_K];[b])\right\} \qquad (4)$$

where

$$d_F(\Sigma(i),\alpha_1(i),\ldots,\alpha_K(i);\beta(i)) =$$

$$= \begin{cases} d(\alpha_j(i),\phi), j \in \{1,\ldots,K\}, \\ d(f(\Sigma(i),\alpha_1(i),\ldots,\alpha_K(i)),\beta(i)),\ \beta(i) \neq \phi. \end{cases} \qquad (5)$$

Example: Let $K = 3$, $g(S,x_1,x_2,x_3) = S + x_1 + x_2 + x_3$, $f = g$, $S_1 = 0$, $E_i = 1$, $1 \leq i \leq 3$, $[a_1] = 10110110001$, $[a_2] = 01011110$, $[a_3] = 101100010110$, and $[b] = 101110$. The edit-sequence and the extended state sequence for a permitted edit-transformation are given by

$$\mathcal{E} = \begin{bmatrix} 1 & \phi & 0 & \phi & \phi & 1 & 1 & \phi & 0 & 1 & \phi & 1 & \phi & \phi & 0 & 0 & \phi & 0 & 1 \\ \phi & \phi & 0 & 1 & \phi & 0 & \phi & \phi & 1 & \phi & \phi & 1 & 1 & \phi & 1 & \phi & \phi & 0 & \phi \\ \phi & 1 & 0 & \phi & 1 & 1 & \phi & 0 & 0 & \phi & 0 & 1 & \phi & 0 & 1 & \phi & 1 & 0 & \phi \\ \phi & \phi & 1 & \phi & \phi & 0 & \phi & \phi & 1 & \phi & \phi & 1 & \phi & \phi & 1 & \phi & \phi & 0 & \phi \end{bmatrix},$$

$$[\Sigma_i] = [\ \phi\ \phi\ 0\ \phi\ \phi\ 0\ \phi\ \phi\ 1\ \phi\ \phi\ 0\ \phi\ \phi\ 0\ \phi\ \phi\ 0\ \phi\ ].$$

Assuming that the elementary distances associated with deletions and effective substitutions are all equal to one, by using (4), one can determine that the distance corresponding to this edit-sequence is 17.

En efficient algorithm for calculating the value of the distance measure defined in (4) can be found in [2].

3. DESCRIPTION OF THE ATTACK

Given the output sequence Y, the attack tries to reconstruct the initial states of the LFSRs that produce the subset $\{X_{j_1}, X_{j_2}, \ldots, X_{j_K}\}$, $j_i \in \{1,\ldots,n\}$ of the set of sequences $\{X_1, X_2, \ldots, X_n\}$, according to the actual partition of the function $\mathfrak{F}$. Given such subset, the procedure determines the candidate initial states that could generate it. For every candidate initial state, all the corresponding decimation sequences can be determined by means of the backtracking procedure that reconstructs all the optimal (i.e. of minimum total weight) edit-transformations. These decimation sequences can then be used for the ciphertext-only attacks on the corresponding decimating subgenerators. In such a way, the original problem is reduced to a set of lower-dimension problems.

The attack introduced above is essentially a decision-making procedure. The algorithm accepts one of the two hypotheses, for all the possible initial states $\{\hat{X}_{j_1}, \hat{X}_{j_2}, \ldots, \hat{X}_{j_K}\}$, $j_i \in \{1,\ldots,n\}$ of the corresponding LFSRs where $\hat{X}_{j_i}$ is the initial state of the LFSR that produces the sequence X_{j_i}. These hypotheses are:

H_0 - the observed sequence Y is produced by the initial states $\{\hat{X}_{j_1}, \hat{X}_{j_2}, \ldots, \hat{X}_{j_K}\}$;

H_1 - the observed sequence Y is not produced by the initial states $\{\hat{X}_{j_1}, \hat{X}_{j_2}, \ldots, \hat{X}_{j_K}\}$.

The decision-making procedure includes the decision threshold T. The value of this threshold can be calculated from the probabilities of 'missing the event' P_m and 'false alarm' P_f. Setting these parameters correctly ensures relatively small number of candidates for the initial states of the investigated LFSRs that could produce the observed output sequence, as well as the high probability that the true solution is in this set of candidates.

The attack proceeds along the following lines:

1. Determine the probability of 'false alarm' P_f and the probability of 'missing the event' P_m, as well as the decision threshold T directly generalizing the method described in [1].
2. Determine the set of solution candidates in the following way: for all the possible states of all the LFSRs $\mathrm{R}_{j_1}, \mathrm{R}_{j_2}, \ldots, \mathrm{R}_{j_K}$, generate the corresponding output sequences of length N and calculate the constrained edit-distance between this set of sequences and the observed output sequence. If this distance is less than the threshold T, put the current initial states of $\mathrm{R}_{j_1}, \mathrm{R}_{j_2}, \ldots, \mathrm{R}_{j_K}$, into the set of solution candidates.
3. For every member of the solution candidates set, reconstruct all the optimal edit-sequences by means of the backtracking procedure. From these edit-sequences, extract the corresponding irregular clocking sequences of the LFSRs $\mathrm{R}_{j_1}, \mathrm{R}_{j_2}, \ldots, \mathrm{R}_{j_K}$.

4. Use the extracted irregular clocking sequences to reconstruct the corresponding initial states of the subgenerators that clock the LFSRs $R_{j_1}, R_{j_2}, \ldots, R_{j_K}$ by means of a ciphertext-only attack adapted to each of the subgenerators.

5. Check whether the obtained solution can generate more than M output bits correctly. The correctness of the solution can be checked measuring the Hamming distance between the generated sequence and the observed sequence. If it is less than the predetermined threshold Q_0, accept the solution. Otherwise, choose the next optimal edit-sequence reconstructed in the step 3. If none of these edit-sequences produces the satisfactory output sequence, then choose another solution candidate determined in the step 2.

4. THE ATTACK ON SOME PARTICULAR SCHEMES

4.1 Consider first the simplest case without the function $\mathfrak{F}$, where the generator is made up of the LFSR R_1 clocked irregularly by the subgenerator consisting only of another LFSR R_{SG} (Fig. 2).

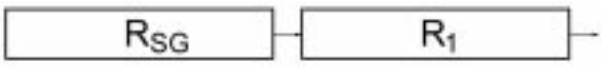

Fig. 2 - The simplest scheme containing two LFSRs

Then, the attack described in the Section 3 has the following particular form:

1. Proceed along the same lines as described in the Section 3, Item 1.

2. Determine the set of solution candidates in the following way: for every possible state of the LFSR R_1, generate the corresponding output sequence of length N and calculate the constrained edit-distance between this sequence and the observed output sequence. The algorithm for calculating the edit-distance for this particular case is given in [1]. The implementation of this algorithm can be carried out in linear space, using only two vectors that correspond to the current and previous substitutions [1]. In order to reconstruct one optimal edit-sequence in linear space, any of the algorithms for reconstructing the longest common subsequence (see, for example, [3]) adapted appropriately can be used.
   If the distance calculated by this algorithm is less than the threshold T, put the current initial state of R_1 into the set of solution candidates.

3. For every member of the solution candidates set, reconstruct all the optimal edit-sequences by means of any classical search method (depth-first or breadth-first). The depth-first search method consumes less memory but is slower. The memory that consumes the breadth-first search method depends on the number of optimal edit-sequences to be reconstructed, which is generally large. This makes the breadth-first search method inconvenient for processing long sequences. The depth-first search method can be implemented using either of the following backtracking procedures:
   Algorithm a: This algorithm uses the matrix of partial constrained edit-distances [1]. This matrix is swept from the right-most column to the left-most column updating the stack of branching points. Upon arrival to the left-most column, the algorithm backtracks to the last branching point, selecting the new direction and updating the stack appropriately.
   The total number of possible optimal paths in the matrix depends on the sequences and cannot be calculated precisely. In [5] the expression for the total number of edit-sequences without constraints is given. Let n_p be the total number of paths in the matrix of partial constrained edit-distances. Then the expression given in [5] can be refined easily if $E_1 = 1$. Therefore the following expression gives the total number of paths in the matrix (optimal and suboptimal) in that case:

$$n_p = \sum_{i=0}^{1} \binom{M}{N-M-i} \tag{6}$$

   Hence, the worst-case time complexity of the Algorithm a) is proportional to Mn_p. The space complexity of this algorithm is proportional to $M(N-M)$.
   Algorithm b: This algorithm uses the vectors needed for calculating the constrained edit-distance in linear space (see item 2). For every step of the algorithm a), the algorithm given in [3] is run once, up to the point equivalent to the right-most column of the corresponding matrix above, then to the point equivalent to the right-most column-1,..., updating the stack of branching points. Upon execution of the algorithm given in [3] up to the point equivalent to the left-most column of the corresponding matrix, the backtracking is performed to the last branching point, selecting the new direction and updating the stack appropriately. The total number of possible paths (optimal and suboptimal) to be traversed in the worst case is given by (6). As for the number of possible optimal paths, the same discussion is valid as that given in the description of the Algorithm a). Thus, the time complexity of the Algorithm b) is proportional to $M(N-M)n_p$.

However, the space complexity of this algorithm is proportional to $N - M$.
Keeping in mind the time and space complexities of the algorithms a) and b), it can be said that the Algorithm b) should be used only when the sequences are too long to maintain in memory the matrix of partial constrained edit-distances.
From the reconstructed optimal edit-sequences, extract the corresponding irregular clocking sequences of the LFSR R_{SG}.

4. Use the extracted irregular clocking sequences to reconstruct the corresponding initial states of R_{SG} by means of the Berlekamp-Massey algorithm [4].

5. Check whether the obtained solution can generate more than M output bits correctly. The correctness of the solution can be checked measuring the Hamming distance between the generated sequence and the observed sequence. If it is less than the predetermined threshold Q_0, accept the solution. Otherwise, choose the next optimal edit-sequence reconstructed in the step 3. If none of these edit-sequences produces the satisfactory output sequence, then choose another solution candidate determined in the step 2.

4.2 Consider now the scheme that contains several blocks. Each of them consists of two LFSRs R_1 and R_2 irregularly clocked by distinct subgenerators based only on single LFSRs R_{1SG}, R_{2SG}, whose outputs are combined in the JK flip-flop (Fig. 3).

In this case, for every given block $f^j = x_1^j \oplus x_2^j \oplus x_1^j x_2^j$, $g^j = f^j$, $g^{-} = 0$, $K = 2$, $L = 1$. The blocks are combined in a balanced non-linear function without memory. The procedure presented above can also be implemented here with the appropriate constrained edit-distance function that corresponds to the transformation of a pair of sequences into an observed sequence via the JK flip-flop. In such a way, the complexity of the cryptanalytic problem would be reduced to the complexity of the reconstruction of the initial states of the pairs of the LFSRs that serve as inputs to the distinct JK flip-flops. But the time complexity of the algorithm for calculating the constrained edit-distance is now proportional to $M(N_1 - M)(N_2 - M)$, where N_1 and N_2 are the respective lengths of the output sequences of the LFSRs R_1 and R_2 without irregular clocking and M is the length of the observed output sequence. The space complexity of this algorithm is proportional to $(N_1 - M)(N_2 - M)$.

5. CONCLUSION

In this paper, an attack on some particular pseudorandom sequence generators containing irregularly clocked LFSRs is described. For every such generator, the time and space complexities of the particular implementation of the general attack have been analysed. Other similar and even more complicated schemes can be cryptanalysed by this method in reasonable time and space.

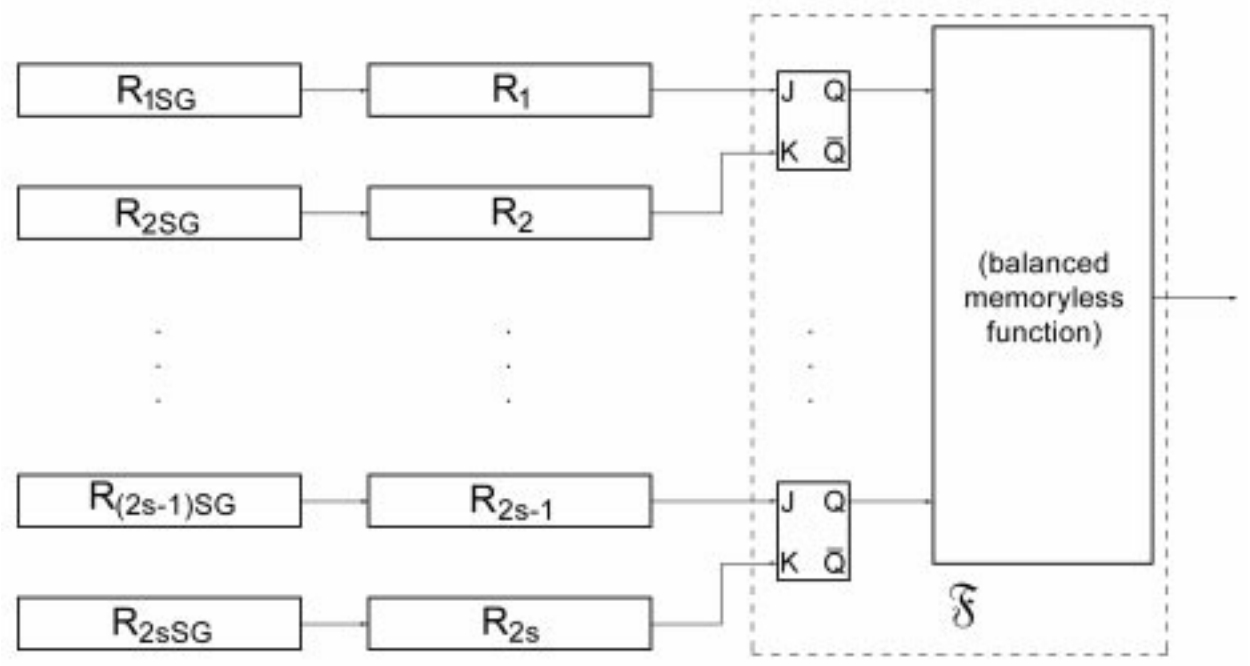

Fig. 3 - The generator containing JK flip-flops

6. REFERENCES

[1] J. Đ. Golić and M.J. Mihaljević, "A Generalized Correlation Attack on a Class of Stream Ciphers Based on the Levenshtein Distance", Journal of Cryptology, Vol. 3, No. 3, 1991, pp. 201-212.

[2] J. Đ. Golić and S.V. Petrović, "Correlation Attacks on Clock-Controlled Shift Registers in Keystream Generators", IEEE Trans. on Computers, Vol. 45, No. 4, 1996, pp. 482-486.

[3] D. Hirschberg, "Serial Computation of Levenshtein Distances", in Pattern Matching Algorithms (A. Apostolico, Z. Galil (Eds.)), Oxford University Press, 1997.

[4] J. Massey, "Shift-Register Synthesis and BCH Decoding", IEEE Trans. on Info. Theory, Vol. IT-15, No. 1, pp. 122-127, 1969.

[5] B.J. Oommen, "Recognition of Noisy Subsequences Using Constrained Edit-Distances", IEEE Trans. Pattern Anal. Mach. Intell. PAMI-9(5), 1987, pp. 676-687.